WHAT IS CLAIMED IS:

1.   A method comprising:

creating first and second data volumes, wherein the first data volume is
unrelated to the second data volume;

refreshing the second data volume to the data contents of the first data volume
so that the second data becomes a point-in-time (PIT) copy of the first
data volume, wherein refreshing the second data volume comprises
overwriting all data of the second data volume with data copied from
the first data volume.

2.   The method of claim 1 further comprising modifying data of the first
data volume before any or all data of the second data volume is overwritten with data
copied from the first data volume.

3.   The method of claim 1 further comprising modifying data of the
second data volume before any or all data of the second data volume is overwritten
with data copied from the first data volume.

4.   The method of claim 1 further comprising creating one or more PIT
copies of the first data volume prior to refreshing the second data volume to the data
contents of the first data volume.

5.   The method of claim 4 wherein one of the PIT copies of the first data
volume is in the virtual state when the second data volume is refreshed to the contents
of the first data volume.

6.   The method of claim 1 further comprising creating one or more PIT
copies of the second data volume prior to refreshing the second data volume to the
data contents of the first data volume.

7.   The method of claim 6 wherein one of the PIT copies of the second
data volume is in the virtual state when the second data volume is refreshed to the
contents of the first data volume.

8.   The method of claim 1 wherein the first data volume is a real or virtual PIT copy of another data volume when the second data volume is refreshed to the contents of the first data volume.

9.   The method of claim 1 wherein the second data volume is a real or virtual PIT copy of another data volume when the second data volume is refreshed to the contents of the first data volume.

10.   The method of claim 1 wherein refreshing the second data volume further comprises generating first and second maps in memory, wherein each of the first and second maps comprises a plurality of entries, wherein each entry of the first map corresponds to a respective memory block that stores data of the first data volume, and wherein each entry of the second map corresponds to a respective memory block that stores data of the second data volume.

11.   The method of claim 10 wherein refreshing the second data volume further comprises:

setting a first bit in each entry of the first map, wherein each first bit of the first map is set to indicate its respective memory block stores valid data;

clearing a first bit in each entry of the second map, wherein each first bit of the second map is set to indicate its respective memory block stores invalid data.

12.   The method of claim 11 further comprising:

setting or clearing a second bit in each entry of the second map to indicate that its respective memory block stores data needed for a PIT copy of the second data volume.

13.   The method of claim 1 further comprising creating a PIT copy of the second data volume before or while refreshing the second data volume to the data contents of the first data volume.

14.   A method comprising:

creating a first hierarchy of data volumes, wherein the first hierarchy
comprises a first primary data volume, wherein each data volume in
the first hierarchy, other than the first primary data volume, is a PIT
copy of another data volume in the first hierarchy or a PIT copy of the
first primary data volume;

creating a second hierarchy of data volumes, wherein the second hierarchy
comprises a second primary data volume, wherein each data volume in
the second hierarchy, other than the second primary data volume, is a
PIT copy of another data volume in the second hierarchy or a PIT copy
of the second primary data volume;

refreshing a data volume of the second hierarchy to the data contents of a data
volume of the first hierarchy.

15.   A computer readable medium storing instructions executable by a
computer system, wherein the computer system implements a method in response to
executing the instructions, the method comprising refreshing a second data volume to
the data contents of a first data volume so that the second data becomes a PIT copy of
the first data volume, wherein refreshing the second data volume comprises
overwriting all data of the second data volume with data copied from the first data
volume, and wherein the first data volume is unrelated to the second data volume
prior to refreshing the second data volume to the data contents of the first data
volume.

16.   The computer readable medium of claim 15 further comprising
modifying data of the first data volume before any or all data of the second data
volume is overwritten with data copied from the first data volume .

17.   The computer readable medium of claim 15 wherein the method
further comprises modifying data copied to the second data volume before any or all
data of the second data volume is overwritten with data copied from the first data
volume.

18.   The computer readable medium of claim 15 wherein the method further comprises creating one or more PIT copies of the first data volume prior to refreshing the second data volume to the data contents of the first data volume.

19.   The computer readable medium of claim 18 wherein one of the PIT copies of the first data volume is in the virtual state when the second data volume is refreshed to the contents of the first data volume.

20.   The computer readable medium of claim 15 wherein the method further comprises creating one or more PIT copies of the second data volume prior to refreshing the second data volume to the data contents of the first data volume.

21.   The computer readable medium of claim 20 wherein one of the PIT copies of the second data volume is in the virtual state when the second data volume is refreshed to the contents of the first data volume.

22.   The computer readable medium of claim 15 wherein the first data volume is a real or virtual PIT copy of another data volume when the second data volume is refreshed to the contents of the first data volume.

23.   The computer readable medium of claim 24 wherein the second data volume is a real or virtual PIT copy of another data volume when the second data volume is refreshed to the contents of the first data volume.

24.   The computer readable medium of claim 15 wherein refreshing the second data volume further comprises generating first and second maps in memory, wherein each of the first and second maps comprises a plurality of entries, wherein each entry of the first map corresponds to a respective memory block that stores data of the first data volume, and wherein each entry of the second map corresponds to a respective memory block that stores data of the second data volume.

25.   The computer readable medium of claim 24 wherein refreshing the second data volume further comprises:

clearing a first bit in each entry of the first map, wherein each first bit of the first map is set to indicate its respective memory block stores valid data;

setting a first bit in each entry of the second map, wherein each first bit of the second map is set to indicate its respective memory block stores invalid data.

26.   The computer readable medium of claim 15 wherein the method further comprises creating a PIT copy of the second data volume before or while refreshing the second data volume to the data contents of the first data volume.

27.   A computer readable medium storing instructions executable by a computer system, wherein the computer system implements a method in response to executing the instructions, the method comprising:

creating a first hierarchy of data volumes, wherein the first hierarchy comprises a first primary data volume, wherein each data volume in the first hierarchy, other than the first primary data volume, is a PIT copy of another data volume in the first hierarchy or a PIT copy of the first primary data volume;

creating a second hierarchy of data volumes, wherein the second hierarchy comprises a second primary data volume, wherein each data volume in the second hierarchy, other than the second primary data volume, is a PIT copy of another data volume in the second hierarchy or a PIT copy of the second primary data volume;

refreshing a data volume of the second hierarchy to the data contents of a data volume of the first hierarchy.

28.   An apparatus comprising:

one or more memories for storing data volumes;

a circuit for creating a first hierarchy of data volumes and a second hierarchy
        of data volumes, wherein the first hierarchy comprises a first primary
        data volume, wherein each data volume in the first hierarchy, other
        than the first primary data volume, is a PIT copy of another data
        volume in the first hierarchy or a PIT copy of the first primary data
        volume, wherein the second hierarchy comprises a second primary data
        volume, wherein each data volume in the second hierarchy, other than
        the second primary data volume, is a PIT copy of another data volume
        in the second hierarchy or a PIT copy of the second primary data
        volume;

a circuit for refreshing a data volume of the second hierarchy to the data
        contents of a data volume of the first hierarchy.

29.   A computer system comprising:

one or more memories for storing data volumes;

a computer system coupled to the one or more memories;

a memory for storing instructions executable by the computer system, wherein
        the computer system implements a method in response to executing the
        instructions, the method comprising:

creating first and second data volumes in the one or more memories, wherein
        the first data volume is unrelated to the second data volume;

refreshing the second data volume to the data contents of the first data volume
        so that the second data becomes a PIT copy of the first data volume,
        wherein refreshing the second data volume comprises overwriting all
        data of the second data volume with data copied from the first data
        volume;

modifying data of the first data volume before all data of the second data
        volume is overwritten with data copied from the first data volume.